

CHAPTER

8

HTML Frames

When you complete this chapter, you will be able to:

- Understand the benefits and drawbacks of frames
- Understand and use frames syntax
- Customize frame characteristics including scroll bars, borders, and margins
- Understand and use frame targeting and special targeting names
- Design content to fit framesets properly
- Design framesets that accommodate different screen resolutions

The HTML frame elements let you partition the canvas area of the browser into multiple windows called **frames**. Each frame can display a separate, independent HTML document. The use of frames has become a subject of controversy on the Web. In many cases, framed Web sites are poorly designed. They detract from the user's experience with heavy download times and confusing navigation. The judicious use of frames, however, can enhance your Web site, allowing you to consistently display navigation information and content concurrently. Frames can be the right solution for solving specific information problems or for providing large collections of content. This chapter explains how to work with frames to display your information effectively.

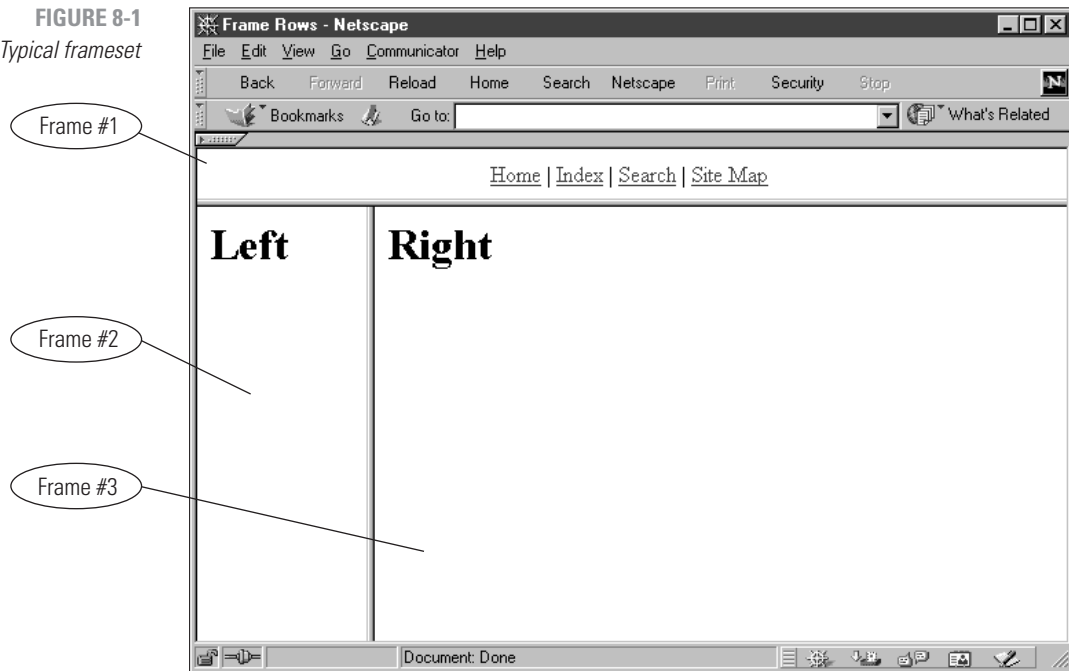


UNDERSTANDING FRAMES

HTML frames were introduced by Netscape for the 2.0 release of its browser. Frames now are supported in a wide variety of browsers. Controversial since their inception, frames can polarize Web designers, some lauding the benefits, others characterizing them as unnecessary. As you will see, frames can work well if you use them correctly. Far too many Web sites use frames just because they can, with no real benefit to the user.

Frames allow you to divide the browser window into independent windows, each displaying a separate HTML document. Figure 8-1 shows an example of a framed set of pages that contains three independent frames.

FIGURE 8-1
Typical frameset

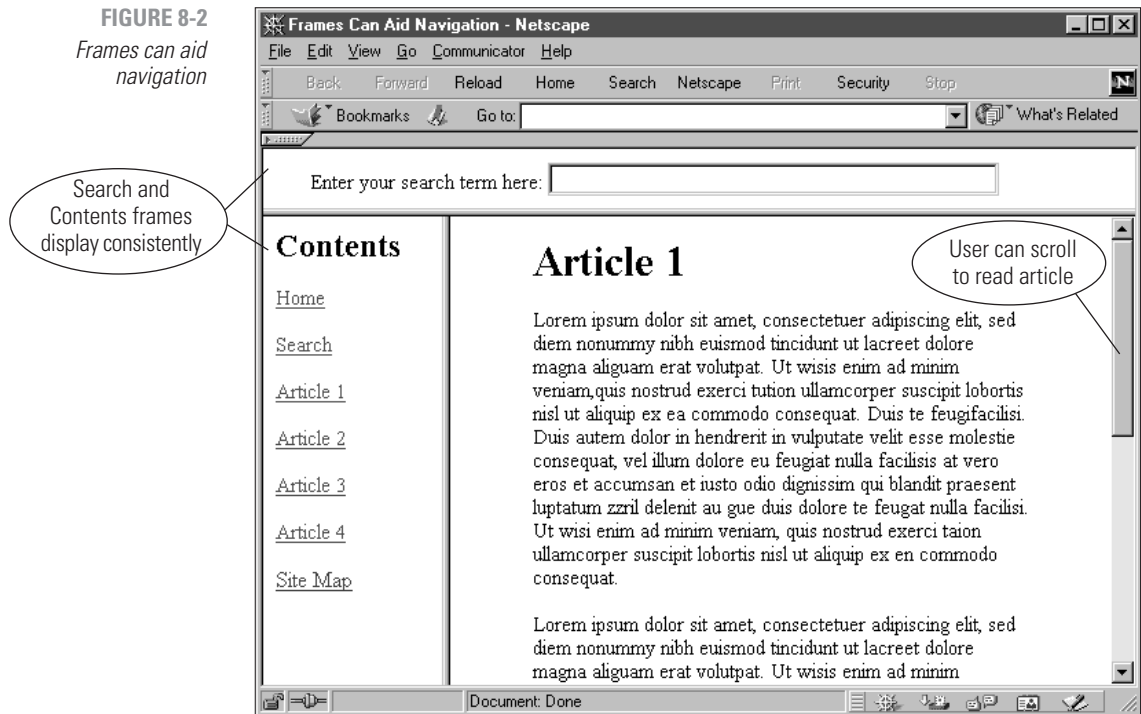


Each of these frames displays a separate HTML file. To create this frameset, first create the individual HTML files, and then create the HTML file containing the frame code that holds the three documents together.

FRAME BENEFITS

Frames offer a number of benefits, including allowing users to scroll independently in one frame without affecting the contents of an adjoining frame. This is an ideal way to present large collections of information that are hard to navigate using the traditional single-page browser display. For example, Figure 8-2 shows a frameset that displays a table of contents in one frame, search tools in another, and content in a third.

FIGURE 8-2
Frames can aid navigation



Because the frames can be scrolled independently, the table of contents and search tool are always visible to the user. This may be exactly the type of additional function you need to solve an information design problem.

Frames work best when you keep them simple. Two or three frames per frame-set should be enough to accommodate your information needs. More frames will clutter the screen, making it difficult for your reader to find information.

FRAME DRAWBACKS

Frames present a variety of drawbacks that you need to consider before committing your content to a frame-based organization.

- **Bookmarks** — Although Internet Explorer 5.0 solves this problem, most browsers do not let the user bookmark individual pages from a Web site. Because the pages are all referenced from a single HTML frames file, the user cannot return to an exact page within a site, only to the main framed page.
- **Download overhead** — Because the browser is loading more than one document, the initial download time can be higher for a framed set of documents than for a single HTML document.
- **Confusing navigation** — Users can become confused if you build complicated framed document sets without enough navigation choices to let users jump to the page of their choice. The Back button in the browser only lets the user load the previous page displayed within the frame, which may not be what the user expects.

TIP

The current versions of both Netscape Navigator and Internet Explorer let you display a framed page in their own browser windows. In Netscape Navigator 4.0, right-click the framed page you want to display and select Open Frame in New Window. In Internet Explorer 4.0 and 5.0, you must select the link that leads to the framed page, which may not always be possible. Right-click the link and select Open in New Window.

- **Visual confusion** — Too many frames within the browser window, each with its own scroll bars, can confuse the user and provide a poor use of the available screen. Limit the number of frames to avoid breaking the browser window into too many sections.
- **Frames and search engines** — Frameset code contains no actual content. The content is contained in the HTML files displayed in the frames. Search engines that read the content of a page for indexing will find no information on a framed page. For this reason it is a good idea to avoid using a framed document as the top-level page for your Web site. Provide a standard HTML page as your top-level page, and use the framed content at a lower level of the Web site. You also can use the `<NOFRAMES>` element (described later in this chapter) to provide content for search engines. For more tips on working with framed pages and search engines, see “Working with Search Engines” in Chapter 9.
- **User preferences** — Users can force their browser to display a page outside of your frameset. Make sure that your pages can stand alone if for some reason a user chooses to display a page on its own. You may want to add a simple text-based navigation bar on each page that is not viewable within the framed document, but can be used to navigate your Web site if a user breaks out of the frameset. You also may want to give the user a choice to navigate your Web site either with or without frames by adding links on the main page to framed and unframed versions of your Web site.

FRAME SYNTAX

HTML frameset documents contain the code that assembles the frames and their contents. Frameset documents themselves have no actual content; therefore they have no `<BODY>` element.

THE `<FRAMESET>` ELEMENT

The `<FRAMESET>` element is the container for the frameset code. The `COLS` and `ROWS` attributes let you specify the characteristics of the frameset. You can specify a frameset as either `COLS` or `ROWS`, but not both. The width (specific to columns) and height (specific to rows) can be expressed as either a percentage value or pixel count in the `COLS` and `ROWS` attributes. As with tables, percentage widths build frames that are relative to the browser window size. Absolute pixel widths are fixed regardless of the browser size. The following code is an example of a simple frameset document that divides the browser canvas into two equal rows:

```
<HTML>
<HEAD>
<TITLE>Frame Rows</TITLE>
</HEAD>
<FRAMESET ROWS="50%,50%">
```

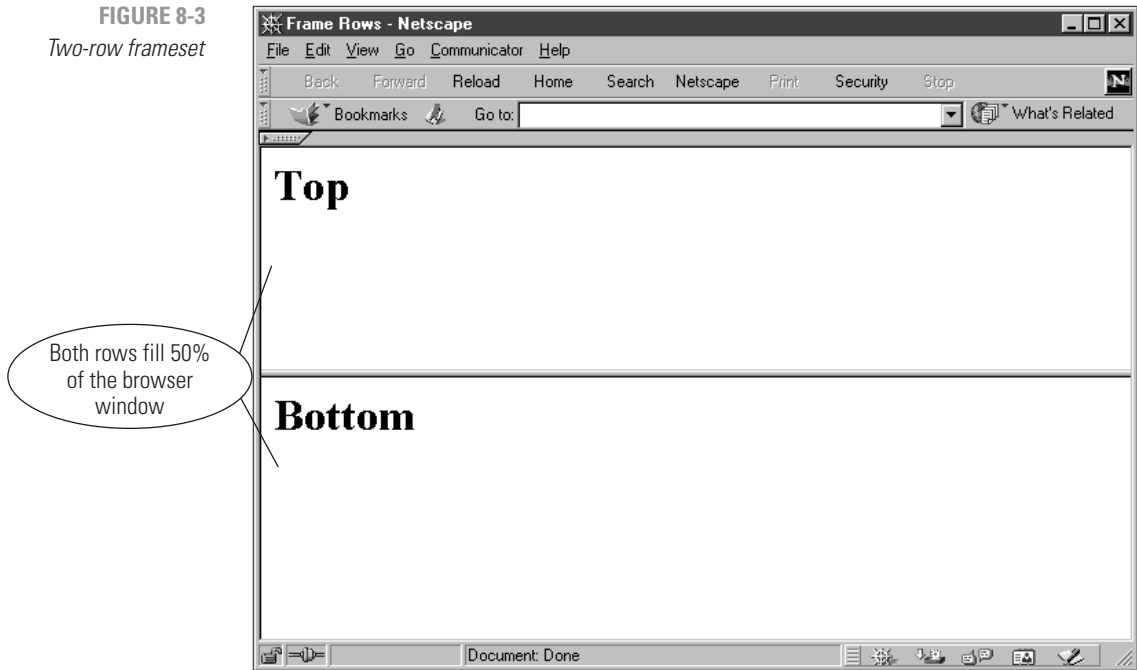
```

<FRAME SRC="top.htm">
<FRAME SRC="bottom.htm">
</FRAMESET>
</HTML>

```

Figure 8-3 shows the result of this code.

FIGURE 8-3
Two-row frameset



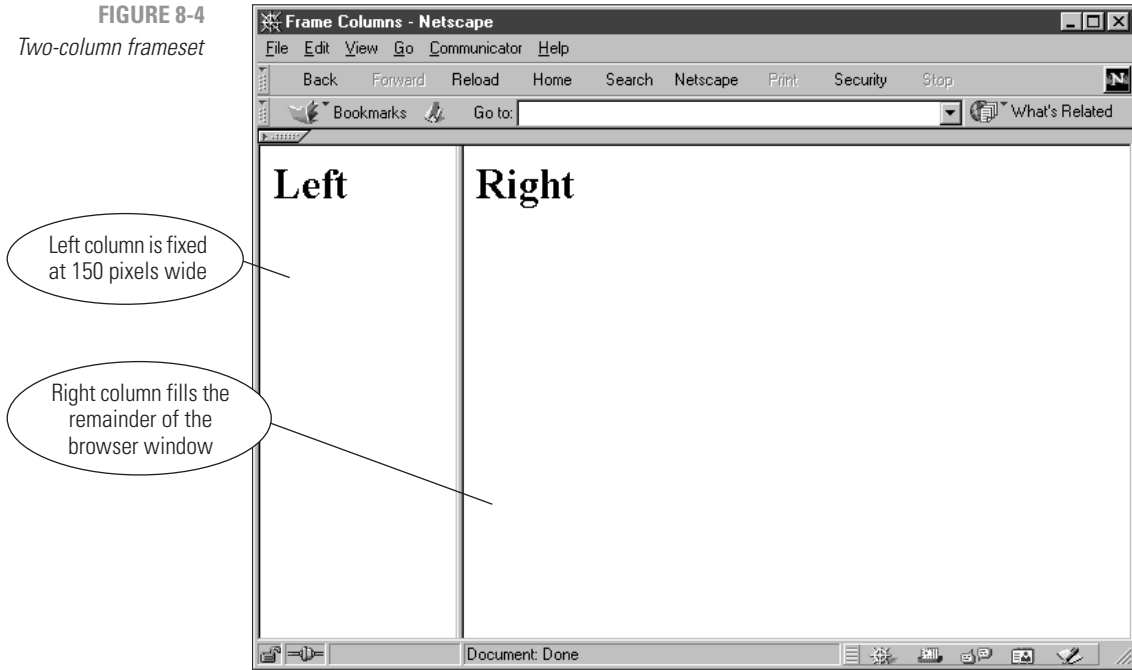
THE <FRAME> ELEMENT

Re-examine the code for Figure 8-3. The <FRAME> element determines the contents of each frame. Row framesets fill top to bottom. The first <FRAME> element tells the browser the file to display in the top row, and the second <FRAME> element points to the file for the bottom row.

<FRAME> is an empty element. The SRC attribute provides the location of the file that displays within the frame. Other attributes to the <FRAME> tag let you name the frame for targeting, specify if frames have a scroll bar, and whether the user can resize the frame. These attributes are described later in this chapter.

In Figure 8-4, the frameset is divided into two column frames.

FIGURE 8-4
Two-column frameset



In this frameset the left column is 150 pixels wide. The right column defaults to the remainder of the browser window size. In the following code, note the syntax in the opening `<FRAMESET>` tag. The `COLS` attribute determines the width of the columns. The first value sets the left column to 150 pixels. The asterisk (*) character tells the browser to fill the right column to the remainder of the browser window.

```
<HTML>
<HEAD>
<TITLE>Frame Columns</TITLE>
</HEAD>
<FRAMESET COLS="150,*">
<FRAME SRC="left.htm">
<FRAME SRC="right.htm">
</FRAMESET>
</HTML>
```

Because the left column is set to a fixed width, it will remain unchanged regardless of the user's browser size or screen resolution. Also note that in column framesets, the frames fill left to right. The first `<FRAME>` element fills the left column, the second fills the right column.

TIP

Even though most browsers can display frames, it still is a good idea to add `<NOFRAMES>` content to help search engines index your framed content. When encountering a framed set of pages, some search engines look for the `<NOFRAMES>` element for content information.

THE `<NOFRAMES>` TAG

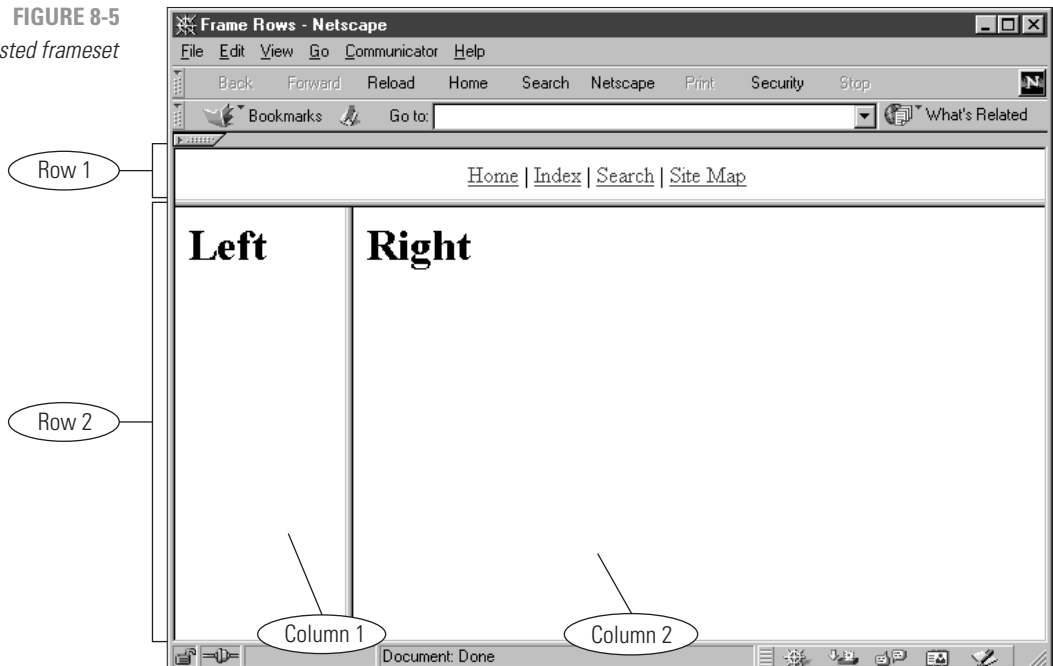
The `<NOFRAMES>` tag lets you provide an alternate page for users who do not have a frames-compliant browser. Enclose the contents of a standard Web page, excluding the `<HTML>` tag, within the `<NOFRAMES>` tag. This alternate page code follows the frameset code in the HTML file:

```
<HTML>
<HEAD>
<TITLE>Frames</TITLE>
</HEAD>
<FRAMESET COLS="75%, 25%">
    <FRAME NAME="index" SRC="index.htm">
    <FRAME NAME="title" SRC="title.htm">
</NOFRAMES>
<BODY>
    (alternate page HTML code)
</BODY>
</NOFRAMES>
</FRAMESET>
</HTML>
```

NESTING FRAMES

Figure 8-5 shows a rows frameset that contains a nested columns frameset in the second row.

FIGURE 8-5
Nested frameset



Nesting allows you to break the screen into both row and column frames. The following code shows the nesting of the framesets:

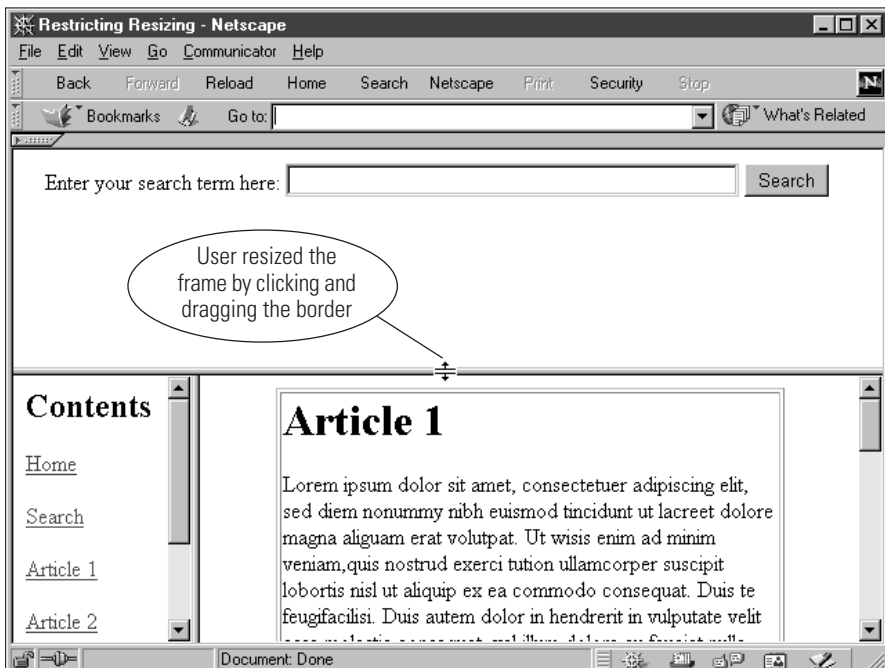
```
<HTML>
<HEAD>
<TITLE>Frame Rows</TITLE>
</HEAD>
<FRAMESET ROWS="40,*">
<FRAME SRC="topnav.htm"> <!-- This is row 1 -->
<FRAMESET COLS="20%,80%"> <!-- The nested frameset fills
the 2nd row -->
<FRAME SRC="left.htm"> <!-- This is column 1 -->
<FRAME SRC="right.htm"> <!-- This is column 2 -->
</FRAMESET>
</FRAMESET>
</HTML>
```

Notice in the above code that two closing `</FRAMESET>` tags are necessary to close both framesets.

RESTRICTING RESIZING

By default, the user has the option of resizing your frames by clicking and dragging the frame border. In most situations you probably want to restrict resizing, so the user sees the frameset the way you intended. Figure 8-6 shows a frameset that the user has resized by clicking and dragging the frame border.

FIGURE 8-6
Default allows the user
to resize the frame



Unless you have a specific reason, you probably want to restrict the user's ability to resize your frames. To restrict resizing, add the `NORESIZE` attribute to the `<SRC>` elements in your frameset as shown in the following code:

```
<HTML>
<HEAD>
<TITLE>Restricting Resizing</TITLE>
</HEAD>
<FRAMESET ROWS="50,*">
<FRAME SRC="search.htm" NORESIZE>
    <FRAMESET COLS="135*">
        <FRAME SRC="contents.htm" NORESIZE>
        <FRAME SRC="article1.htm" NORESIZE>
    </FRAMESET>
</FRAMESET>
</HTML>
```

CONTROLLING SCROLL BARS

By default, scroll bars in frames are set to appear automatically if the content is not accessible within the frame window. In most cases this is the best setting for scroll bars, because you will not need to worry about them. No matter the user's browser size or screen resolution, if scroll bars are necessary they will appear. Sometimes, however, you may want to control whether scroll bars display. Use the `SCROLLING` attribute in the `<SRC>` element to control scroll bars. The valid values are "YES," "NO," or "AUTO," which is the default setting.

Figure 8-7 shows a three-frame frameset. Notice that the top frame displays a scroll bar even though no additional content follows the search text box.

The browser displays a scroll bar because the height of the top frame is slightly smaller than the browser finds necessary to display the contents. One way to solve this problem is to change the height of the frame. Because this frame looks good at this height, however, you can remove the scroll bar by adding the `SCROLLING=NO` attribute to the `<SRC>` element. Figure 8-8 shows the result of the attribute addition.

FIGURE 8-7
*Unnecessary default
 scroll bar*

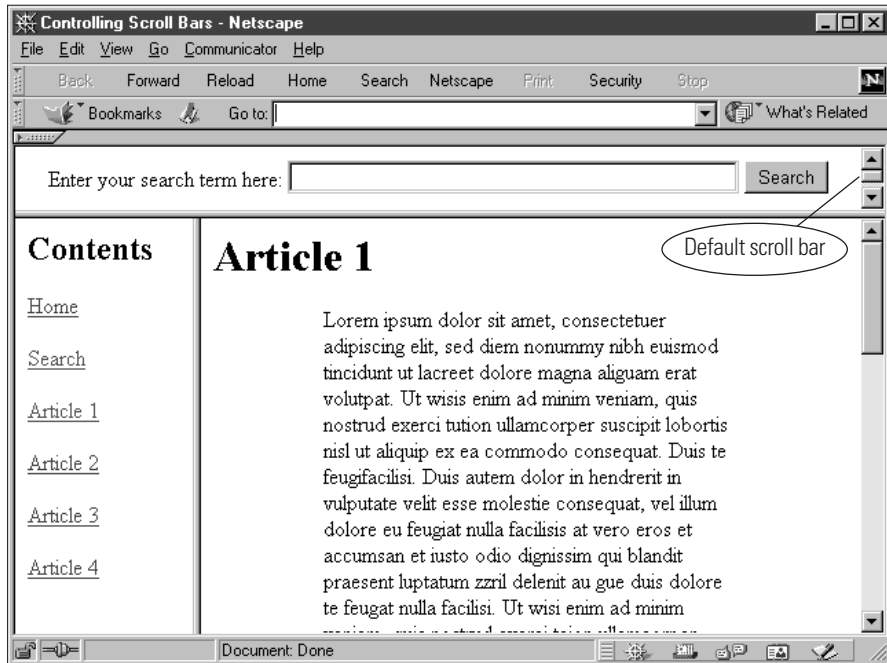
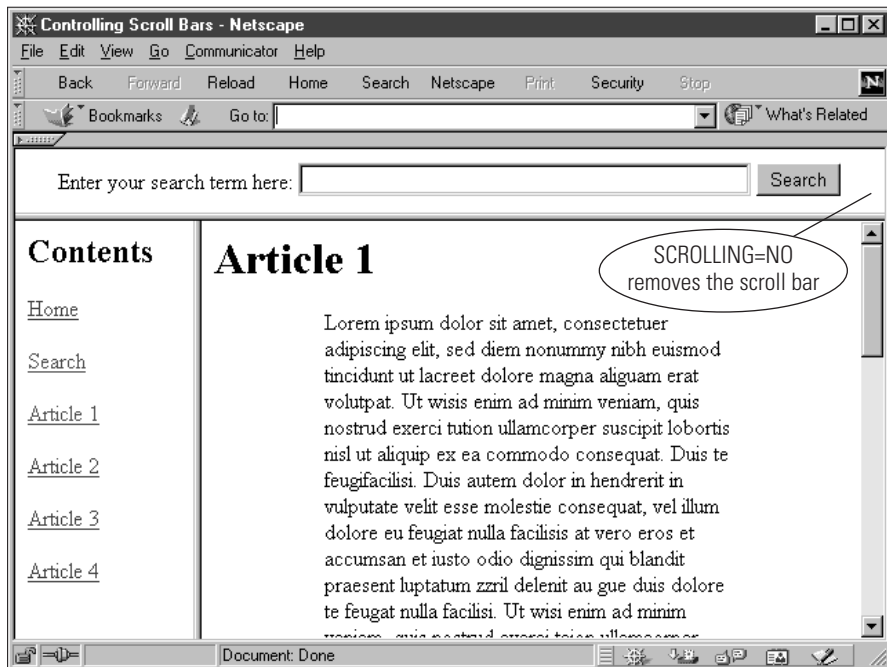


FIGURE 8-8
Frame with no scroll bar



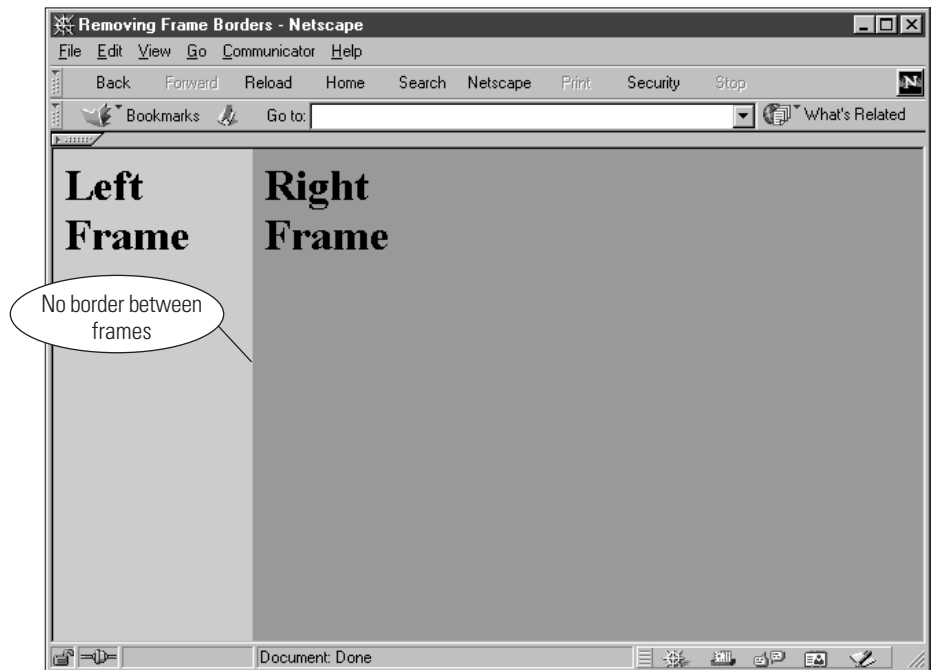
The scroll bar no longer displays, which enhances the look of the frameset. Also, because a scroll bar indicates additional information, omitting the scroll bar reflects more accurately the content of this page. The code for the complete frameset follows:

```
<HTML>
<HEAD>
<TITLE>Controlling Scroll Bars</TITLE>
</HEAD>
<FRAMESET ROWS="50,*">
<FRAME SRC="search.htm" SCROLLING=NO>
<FRAMESET COLS="135,*">
<FRAME SRC="contents.htm">
<FRAME SRC="article1.htm">
</FRAMESET>
</FRAMESET>
</HTML>
```

CONTROLLING FRAME BORDERS

As with tables, you can choose not to display frame borders, or to remove the default border spacing between frames entirely. This technique lets you create seamless frames with no visible dividing line, unless a scroll bar pops up. Figure 8-9 shows an example of a two-frame frameset with the frame borders turned off.

FIGURE 8-9
Frame border turned off



Unfortunately, the two major browsers, Netscape Navigator and Internet Explorer, do not agree on the attributes you should use to achieve this effect. Netscape Navigator 4.0 requires `FRAMEBORDER=NO` and `BORDER=0` attributes. Internet Explorer 5.0 can interpret correctly `FRAMEBORDER=NO`, but you need to set `FRAMESPACING=0` to remove the default border spacing between frames. All of these attributes reside in the opening `<FRAMESET>` tag. To make sure that the frame border and spacing are turned off for both browsers, use the syntax shown in the code for the above frameset:

```
<HTML>
<HEAD>
<TITLE>Removing Frame Borders</TITLE>
</HEAD>
<FRAMESET COLS="150,*" FRAMEBORDER=NO FRAMESPACING=0
BORDER=0>
<FRAME SRC="leftgray.htm">
<FRAME SRC="rightgray.htm" NAME="CONTENT">
</FRAMESET>
</HTML>
```

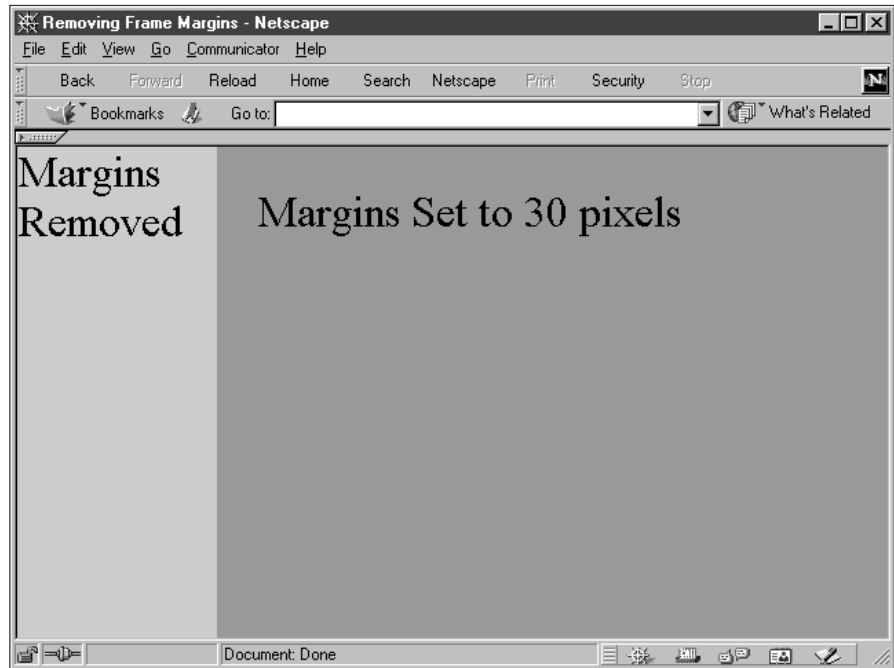
CONTROLLING FRAME MARGINS

Two frame attributes let you control the pixel width of both the vertical and horizontal margins in a frame. `MARGINWIDTH` lets you control the left and right margins, while `MARGINHEIGHT` affects the top and bottom margins. Setting these attributes to zero (0) lets you remove the margins entirely, allowing your content to touch the sides of the frame. You most likely would use these attributes in combination with the frame border attributes described above.

Add the `MARGINHEIGHT` and `MARGINWIDTH` attributes to the `<FRAME>` element for the frame you want to affect. Figure 8-10 shows a frameset with two different margin settings.

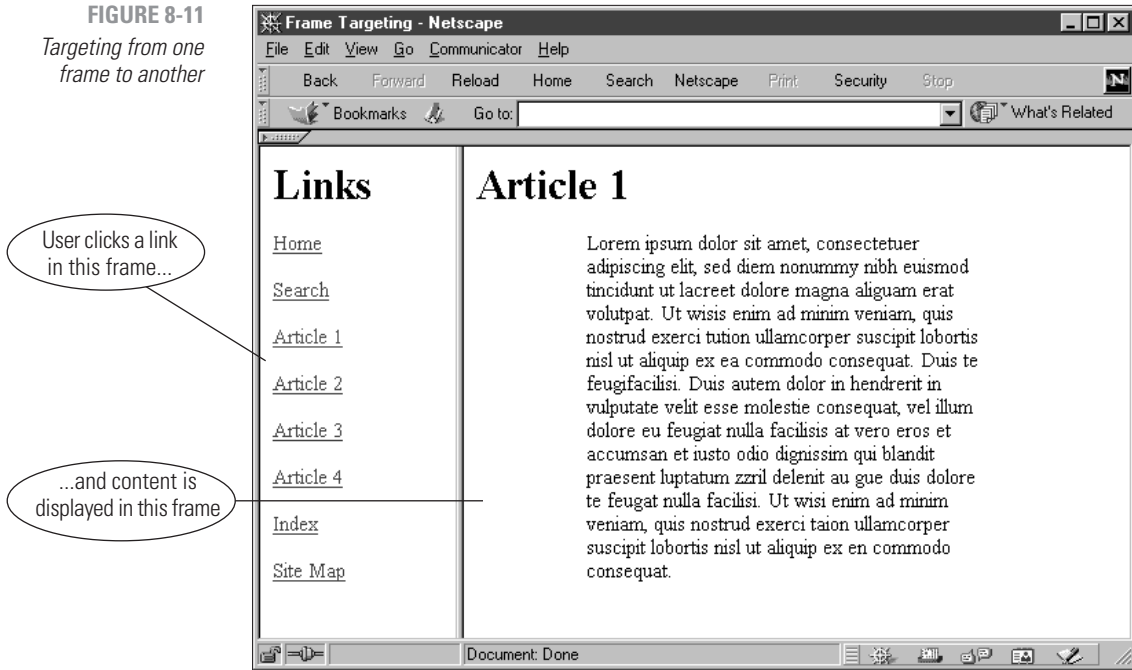
The left frame has both margin attributes set to zero (0). Even so, the top margin still includes some space, due to built-in leading in the line of text. The left margin, however, has been completely removed. In the right column both margins have been set to 30 pixels.

FIGURE 8-10
*Frame borders and
margins removed*



TARGETING IN FRAMESETS

The power of frames comes from the ability to have one frame display information consistently while the contents of a second frame might change based on the user's choice. By default, a link loads into the same frame from which it was selected. You can change this default behavior and target the destination of a link to another frame in the frameset. In this section you will learn how to target within the simple frameset, as shown in Figure 8-11.

FIGURE 8-11*Targeting from one frame to another*

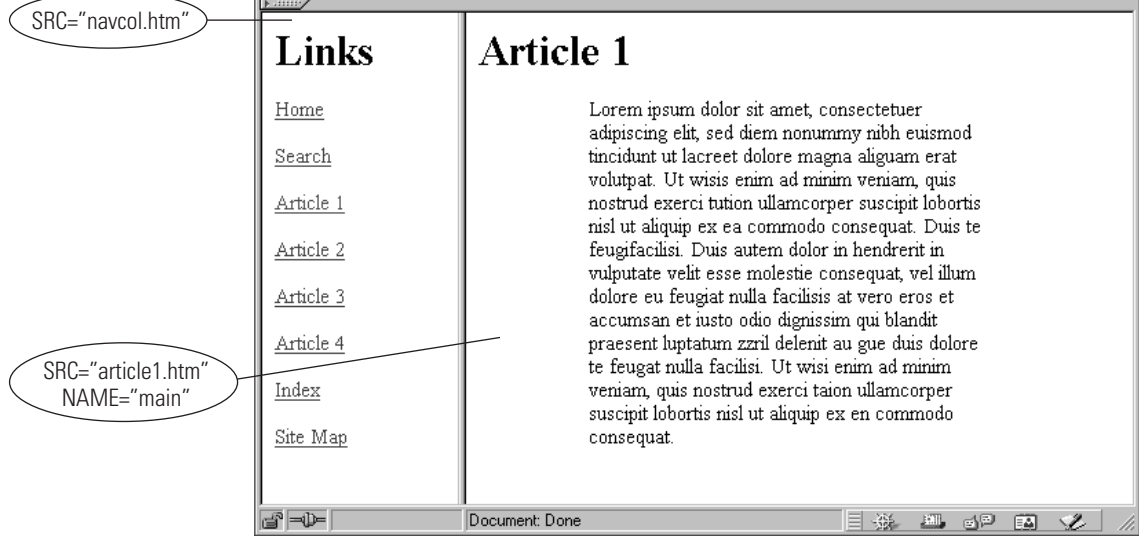
To target from one frame to another, you must perform two tasks:

1. Name your frames using the NAME attribute in the FRAME element.
2. Target links to display their content in the named frame.

NAMING FRAMES

To name a frame, add the NAME attribute to the <FRAME> element. You do not have to name all of the frames within a frameset, only the frames you want to target. Figure 8-12 shows a frameset with two frames. The right frame is named “content”.

FIGURE 8-12
Naming a frame for
targeting



Here is the HTML code for the frameset illustrated in Figure 8-12.

```
<HTML>
<HEAD>
<TITLE>Frame Targeting</TITLE>
</HEAD>
<FRAMESET COLS="150,*">
<FRAME SRC="navcol.htm">
<FRAME SRC="article1.htm" NAME="main">
</FRAMESET>
</HTML>
```

The NAME attribute in the second <FRAME> element names the right frame window “main.” You now can target this window to display linked content.

TARGETING NAMED FRAMES

To target the named frame, you must edit the HTML document that contains the <A> elements and provide TARGET attributes that tell the browser which frame displays the content. You can use the TARGET attribute in either the <BASE> or <A> elements.

Targeting in the <BASE> Element

In this example, the HTML document that occupies the left frame window is named “navcol.htm.” Adding the <BASE> element lets you set the default target

frame for all of the links in the document. <BASE> is an empty tag that resides in the <HEAD> section of the document. The code in navcol.htm shows the <BASE> element.

```
<HTML>
<HEAD>
<TITLE>Frame Links</TITLE>
<BASE TARGET="main">
</HEAD>
<BODY>
<H1>Links</H1>
<P><A HREF="index.htm">Home</A></P>
<P><A HREF="search.htm">Search</A></P>
<P><A HREF="article1.htm">Article 1</A></P>
<P><A HREF="article2.htm">Article 2</A></P>
<P><A HREF="article3.htm">Article 3</A></P>
<P><A HREF="article4.htm">Article 4</A></P>
<P><A HREF="contindx.htm">Index</A></P>
<P><A HREF="sitemap.htm">Site Map</A></P>
</BODY>
</HTML>
```

Notice the <BASE> element contains the TARGET attribute set to “content.” This establishes the default window target name for all of the links contained in the file. Any link that the user selects will display in the frame window named “content.”

TIP

Make sure that you match the case of both the TARGET and NAME attributes, or the browser will not be able to resolve the target name.

Targeting in the <A> element

You can override a default base target by using the TARGET attribute in the <A> element. This allows you to target a specific link to a destination different from the base target. You can target a different window within the frameset, or can use one of the special targeting values in the next section. The following <A> element targets article1.htm to the frame named “frame2.”

```
<A HREF="article1.htm" TARGET="frame2">Article 1</A>
```

USING SPECIAL TARGET NAMES

There are four special target names that you can use with the TARGET attribute in either the <BASE> or <A> elements. Table 8-1 lists the special names.

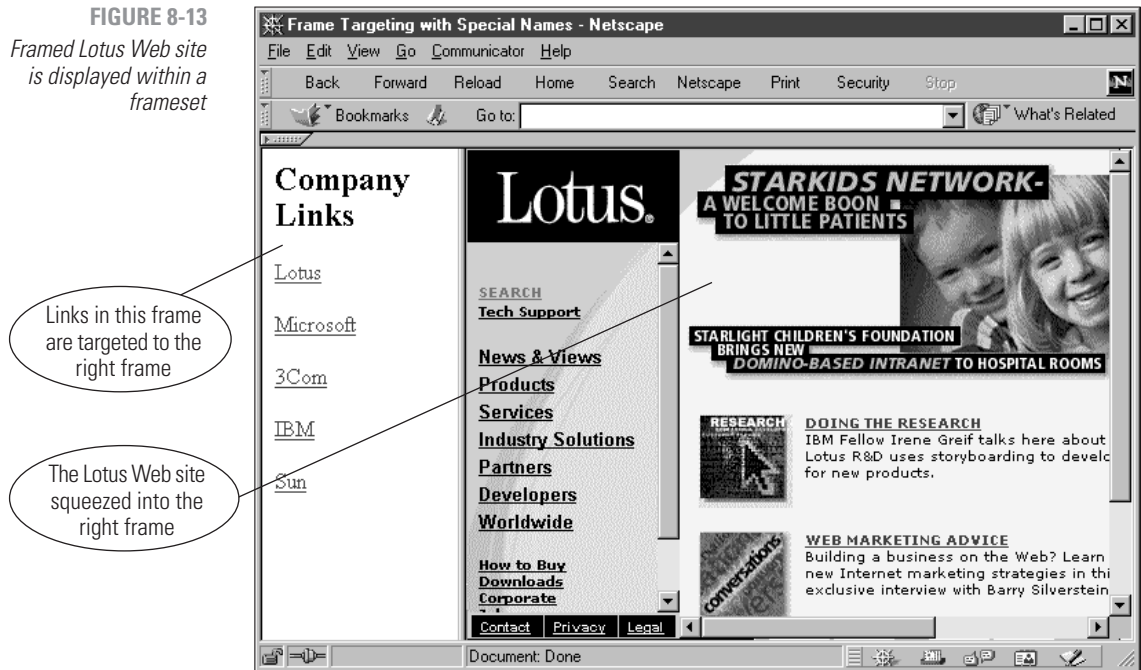
Notice that all of these special names begin with an underscore. Any other target name that begins with an underscore will be ignored by the browser.

Special target names can help you in a variety of situations. For example, you can use special target names when you link to other sites from within a frameset. Figure 8-13 shows the Lotus Web site displayed in a frame.

TABLE 8-1
Special target names

Name	Description
<code>_self</code>	This is the default behavior for links in a frameset. The linked content is loaded into the same window as the <A> element. You most likely would use this in the <A> element to override a base target.
<code>_blank</code>	This name opens a new browser window to display the linked content. This result can cause navigation confusion for the users, who may not realize that they are looking at a new instance of the browser.
<code>_parent</code>	This name lets you break out of a child frameset and display the link in the parent frameset one level up in the frameset hierarchy. This name only is useful when you have a link in a frameset that displays an embedded frameset. In most cases using embedded framesets is poor navigation design that will confuse your user.
<code>_top</code>	The most useful of all the special names, <code>_top</code> lets you remove frames and display the linked content in a fresh browser window.

FIGURE 8-13
Framed Lotus Web site is displayed within a frameset



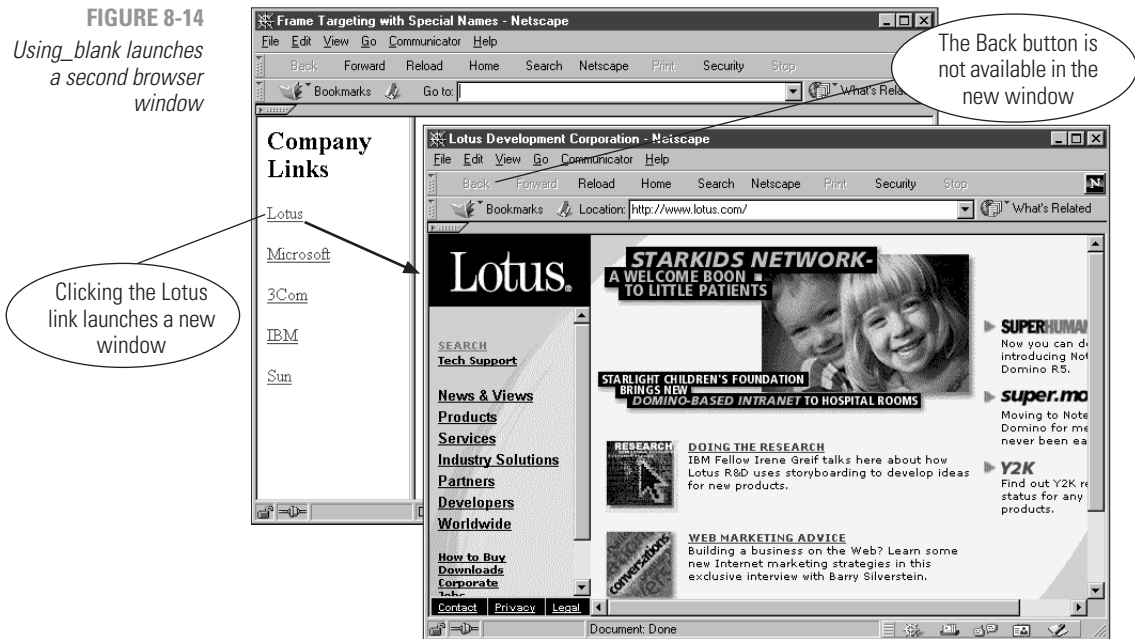
As you can see, loading the framed Lotus Web site within a frameset does not work well. The size of the frame is too small to display the Web site properly, causing multiple horizontal and vertical scroll bars. You can solve this problem by

using special target names to break out of the frameset. There are two ways to handle this problem:

- Use `_blank` to load the Lotus Web page in a new browser window
- Use `_top` to load the Lotus Web page to the top of the existing window

Using `_blank`

The `_blank` special target name lets you load the linked content into a new instance of Netscape Navigator. Figure 8-14 shows the results.



Notice that the Back button is not available in the new browser window because this is the first page in the new window. Not being able to use Back can be disorienting to users who rely on it for navigation.

Remember that the targeting information is in the file that contains the links, not the frameset file. In this example, the `_blank` target name resides in the `<BASE>` element, setting the default target for all of the links within the file. The following code shows the use of `_blank` in the left navigation column in Figure 8-14 above.

```
<HTML>
<HEAD>
<TITLE></TITLE>
</HEAD>
<BASE TARGET="_blank">
<BODY>
<H2>COMPANY LINKS</H2>
<P><A HREF="http://www.lotus.com">Lotus</A></P>
```

```

<P><A HREF="http://www.microsoft.com">Microsoft</A></P>
<P><A HREF="http://www.3com.com">3COM</A></P>
<P><A HREF="http://www.ibm.com">IBM</A></P>
<P><A HREF="http://www.sun.com">SUN</A></P>
</BODY>
</HTML>

```

TIP

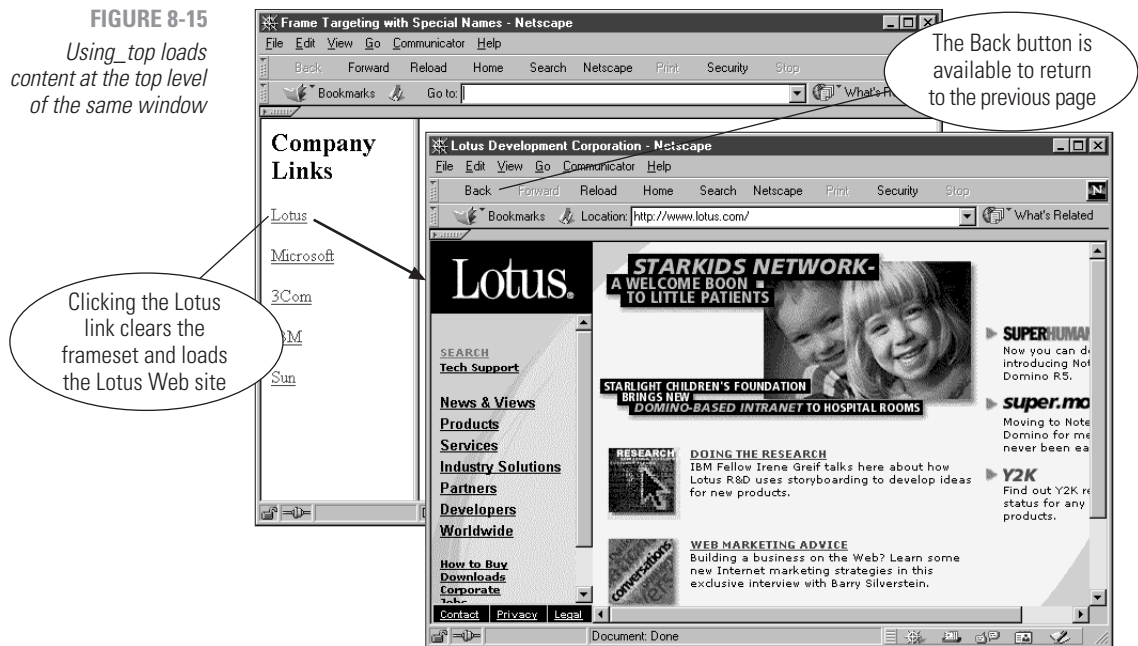
If you decide to use `_blank`, you can help the user by letting them know that clicking the link will open a new browser window.

This code contains one flaw. Using `_blank` as the default target name means that every link in this window will launch a new browser window. Before long, the user's computer either will run out of memory or the screen will become cluttered by overlapping windows. For this reason, limit the use of `_blank` for special purposes, or do not use it at all.

Using `_top`

Using `_top` as a special target name displays the linked content in a non-framed window using the same instance of the browser. Figure 8-15 shows the results.

FIGURE 8-15
Using `_top` loads content at the top level of the same window



The browser clears the frameset and loads the Lotus link in the same window. The Back button is available if the user wants to return to the previous page. Because the browser maintains only one open window, there is no additional memory overhead or confusion for the user. This effect is hard to reproduce on paper, so use the sample files on the *Principles of Web Design* Companion Web Site to see an example.

PLANNING FRAME CONTENT

The variable nature of the World Wide Web affects your framed pages. If you are planning on building a framed site, you must prepare your content for display within a frameset. Frameset display is affected by the base screen resolution you use to code your framed pages. You must decide on the lowest common denominator screen resolution that will display the frameset effectively.

FRAMES AND SCREEN RESOLUTION

Frame scroll bars and borders add to the screen space requirements of frames. Many Web sites that use frames build them for a base resolution of 800 x 600, forcing users to change their screen resolutions in order to view the content. Even if you decide to code for a higher resolution, you always should test at a 640 x 480 resolution because some users will view your Web site at this resolution. Also, test your work in different browsers. Small differences between the way browsers display frames can affect their look significantly. Figures 8-16 and 8-17 show the Anne Frank House Web site at 640 x 480 in Netscape Navigator 4.0 and Internet Explorer 5.0.

FIGURE 8-16
Anne Frank House Web
site in Netscape
Navigator 4.0

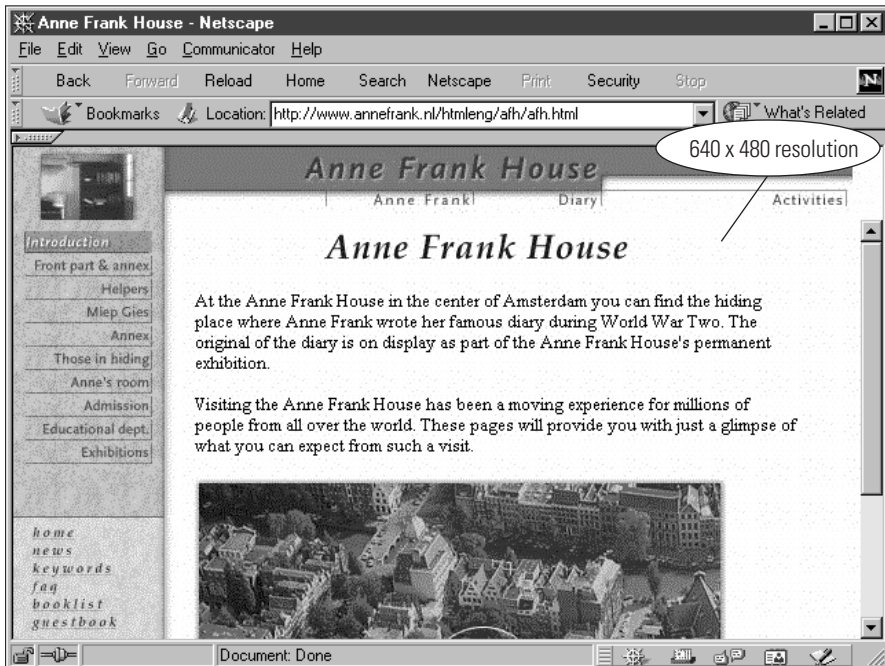
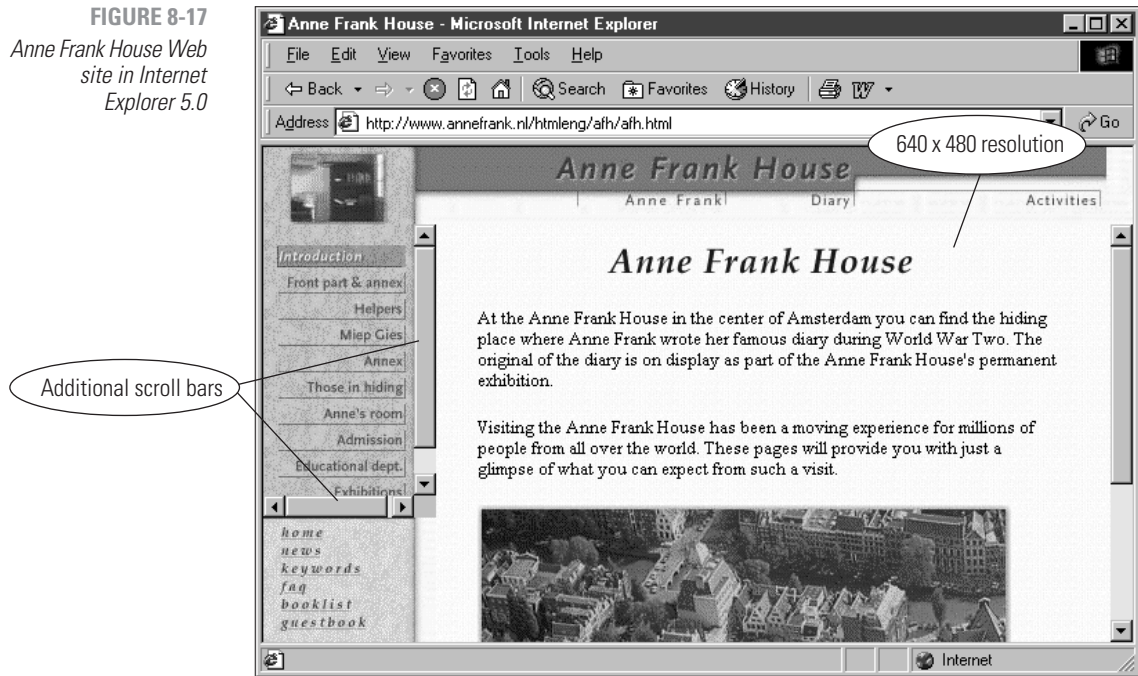


FIGURE 8-17
*Anne Frank House Web
 site in Internet
 Explorer 5.0*



The additional scroll bars in Internet Explorer distract from the seamless nature of the frameset. The HTML author would have to tinker with the frameset measurement values and scroll bar settings to get the frameset to display consistently across both browsers.

DESIGNING EFFECTIVE FRAMES

You must build your pages to fit within the frames in which they will display. You also will need to accommodate different screen resolutions that can affect the size of the frame within the frameset. As with tables, decide whether you will use fixed or relative framesets. You also can choose to mix these two measurement types within a single frameset, which can be the best way to handle multiple screen resolutions.

MIXING FIXED AND VARIABLE FRAMES

Examine the following sample frameset that mixes a fixed frame and a variable frame to accommodate different resolutions.

Here is the code for a sample frameset:

```
<HTML>
<HEAD>
<TITLE>Designing Frame Content</TITLE>
</HEAD>
<FRAMESET COLS="125,*">
```

```

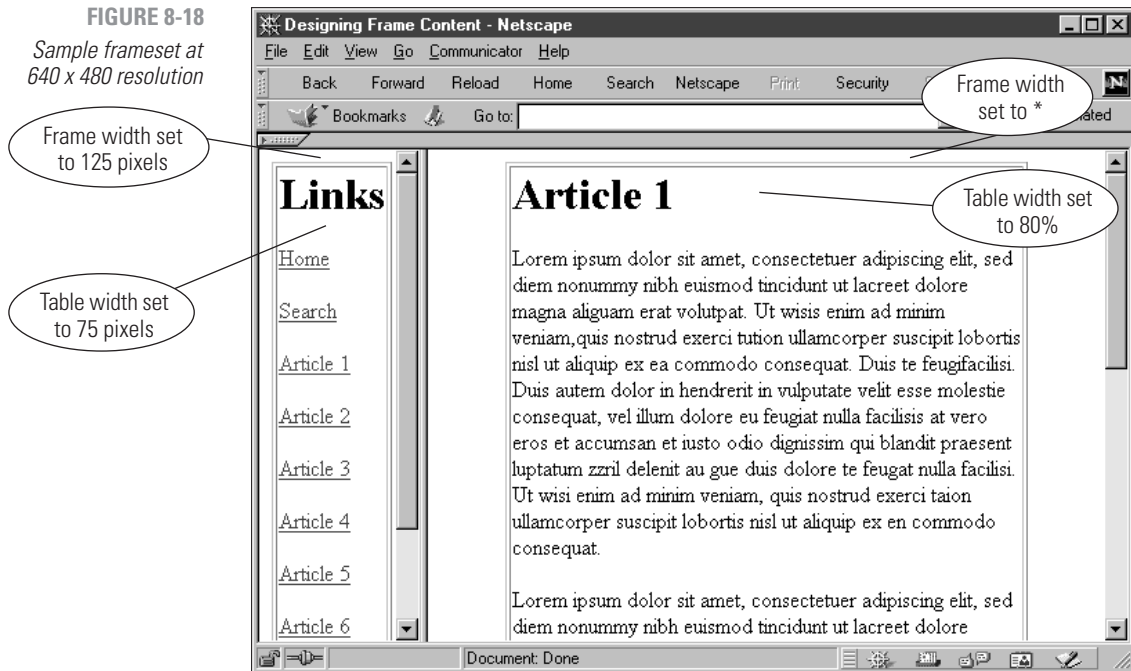
<FRAME SRC="leftfxd.htm" >
<FRAME SRC="article1.htm" name="main">
</FRAMESET>
</HTML>

```

Notice that the code for the left column is fixed at 125 pixels. The asterisk (*) wildcard character sets the right column to a variable width that changes, based on the browser size. Figure 8-18 shows the frameset at 640 x 480 resolution.

FIGURE 8-18

Sample frameset at 640 x 480 resolution



Tables in each HTML file keep the content aligned within the frameset. The table borders are turned on so you can see them. The table in the left frame has a fixed width of 75 pixels. The difference between the frame width of 125 and the table width of 75 is taken up by the scroll bar and frame margins. This frame and content will display consistently regardless of the screen resolution. The code for the table in the left frame content, `leftfxd.htm`, follows:

```

<HTML>
<HEAD>
<TITLE>Left Fixed Column</TITLE>
<BASE TARGET="main">
</HEAD>
<BODY>
<TABLE WIDTH=75 BORDER>
<TR>

```

```

<TD>
<H1>Links</H1>
<A HREF="index.htm">Home</A>
<P><A HREF="search.htm">Search</A></P>
<P><A HREF="article1.htm">Article 1</A></P>
<P><A HREF="article2.htm">Article 2</A></P>
<P><A HREF="article3.htm">Article 3</A></P>
<P><A HREF="article4.htm">Article 4</A></P>
<P><A HREF="article5.htm">Article 5</A></P>
<P><A HREF="article6.htm">Article 6</A></P>
<P><A HREF="contindx.htm">Index</A></P>
<P><A HREF="sitemap.htm">Site Map</A></P>
</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```

The right frame is a variable width; therefore, the content within the right frame is contained in a centered table set to an 80 percent width. The 80 percent width allows for white space on both sides of the text. The code for the right frame content, `article1.htm`, follows:

```

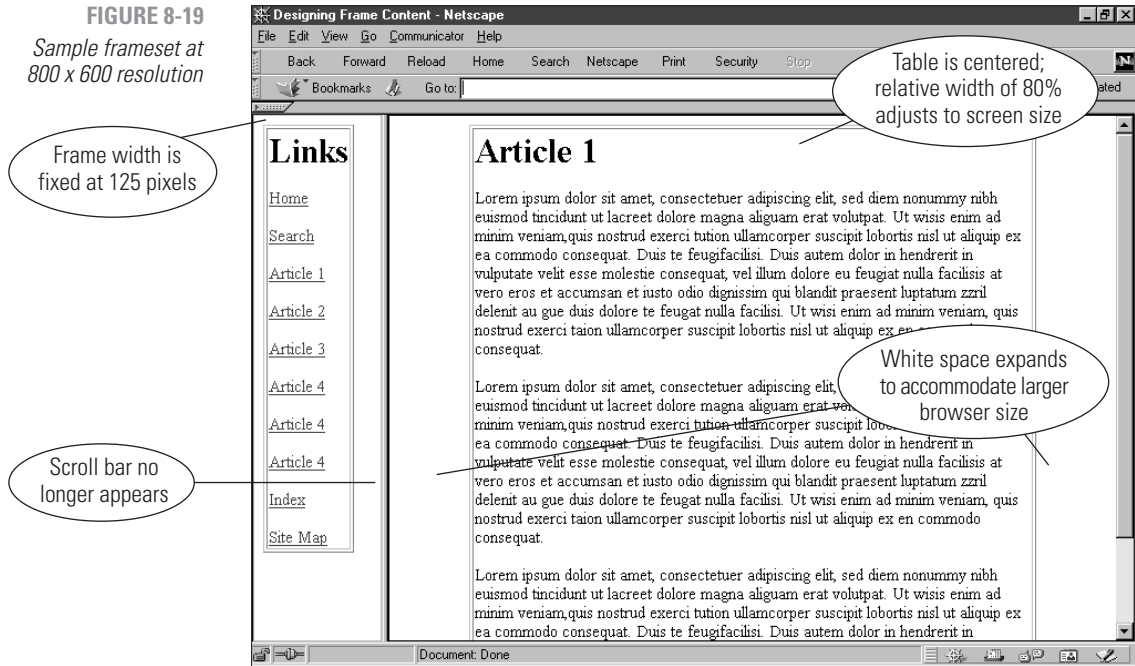
<HTML>
<HEAD>
<TITLE>Right Variable Column</TITLE>
</HEAD>
<BODY>
<DIV ALIGN=CENTER>
<TABLE WIDTH=80% BORDER>
<TR>
<TD>
<H1>Article 1</H1>
<P>Lorem ipsum dolor sit amet, consectetur adipiscing
elit, sed diem nonummy nibh euismod ti . . .(content
abbreviated)</P>
</TD>
</TR>
</TABLE>
</DIV>

```

Now look at the same frameset at an 800 x 600 resolution in Figure 8-19.

FIGURE 8-19

Sample frameset at
800 x 600 resolution



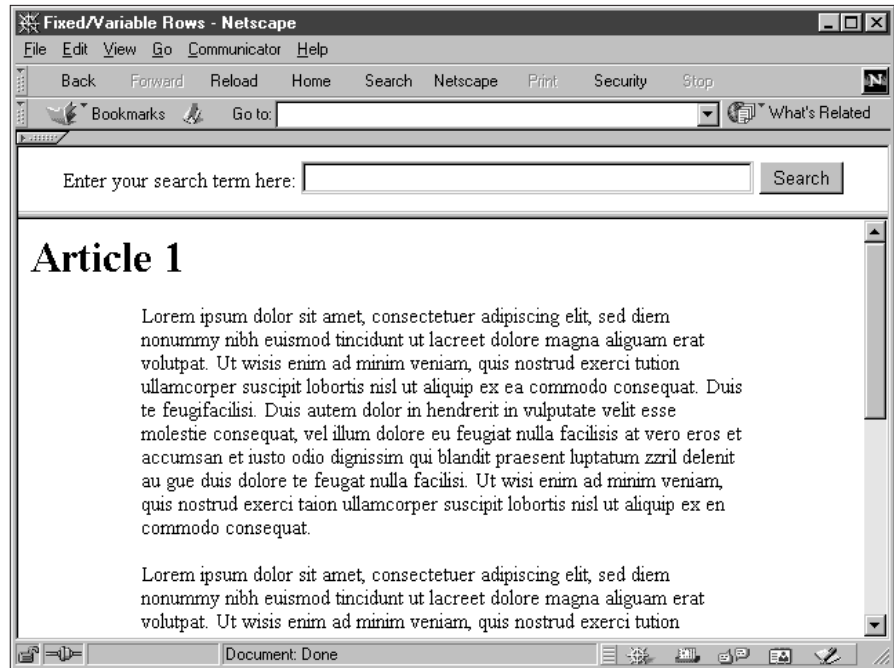
The left frame is fixed at 125 pixels. The only difference is that the scroll bar no longer appears in the left frame because of the increased browser size. The right column still has centered content with white space on both sides of the text. Because the table in the right frame is variable, it adjusts to the new screen resolution.

If you are building a row frameset, you can use this same fixed and variable coding method. Set one row to a fixed width, and let the second row default to the remainder of the browser canvas, as shown in the following code:

```
<HTML>
<HEAD>
<TITLE>Fixed/Variable Rows</TITLE>
</HEAD>
<FRAMESET ROWS="80,*">
<FRAME SRC="topfxd.htm">
<FRAME SRC="btvarble.htm" >
</FRAMESET>
</HTML>
```

This code creates the page shown in Figure 8-20.

FIGURE 8-20
*Fixed top row, variable
 second row*



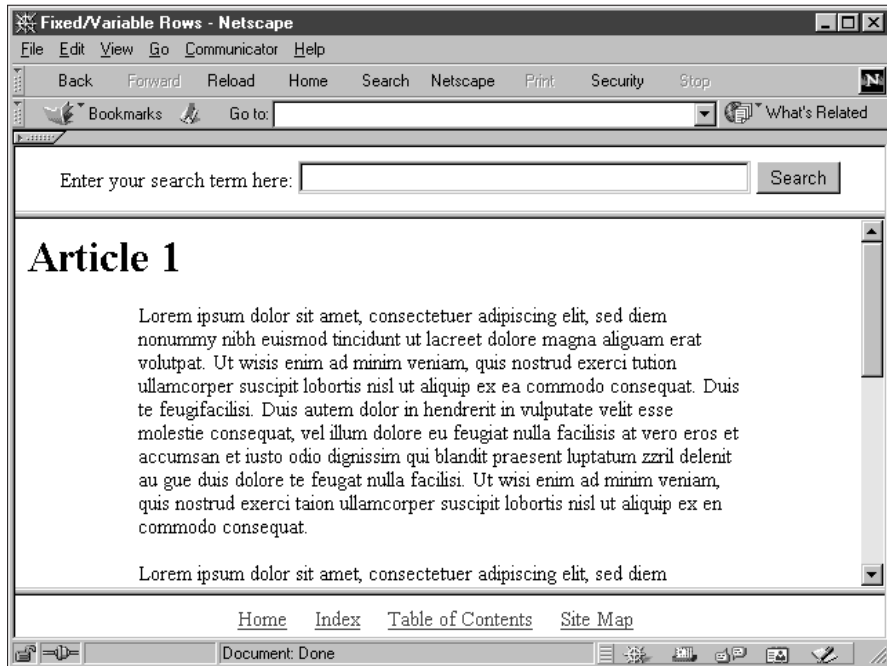
The top row is 80 pixels high. The second row fills the browser window regardless of screen resolution.

A final example of the fixed/variable method is a common three-row frame-set. In this set the top and bottom rows contain navigation, while the middle row contains content. The top and bottom rows are fixed, and the middle row is variable. Here is the code:

```
<HTML>
<HEAD>
<TITLE>Fixed/Variable Rows</TITLE>
</HEAD>
<FRAMESET ROWS="50,*,35">
<FRAME SRC="search.htm" SCROLLING=NO>
<FRAME SRC="article1.htm" NAME="MAIN">
<FRAME SRC="botmcnt.htm" SCROLLING=NO>
</FRAMESET>
</HEAD>
```

Figure 8-21 shows this frameset.

FIGURE 8-21
*Three-row frameset
 with variable
 middle row*



SUMMARY & REVIEW

- Frames allow you to divide the browser window into independent windows, each displaying a separate HTML document. Frames allow users to scroll in one frame without affecting the contents of an adjoining frame. This is an ideal way to present large collections of information that are hard to navigate using the traditional single-page browser display.
- Although frames offer a number of benefits, you should use them judiciously. Make sure that your content benefits from the use of frames, and build simple framesets with no more than two or three frames.
- Frames present drawbacks that you need to consider before committing your content to a frame-based organization. These drawbacks include the inability to bookmark pages, increased download times, potential navigation and visual confusion, and the inability of search engines to find framed documents.
- Because frameset documents contain no content, they are not the best choice for the top-level page of your Web site. Consider using a standard HTML page for the top-level, and then link to your framed content.
- HTML frameset documents contain the code that assembles the frames and their contents. The `<FRAMESET>` element is the container for the frameset code. Frameset documents themselves have no actual content; therefore they have no `<BODY>` element.
- The `<FRAME>` element is an empty element that determines the contents of each frame. The `SRC` attribute provides the location of the file that displays within the frame. Other attributes to the `<FRAME>` tag let you name the frame for targeting, specify whether frames have a scroll bar, and indicate whether the user can resize the frame.

- Use the <NOFRAMES> element to contain alternate information, about your Web site for users with browsers that cannot display frames.
- You can design frames where one frame displays unchanging information while the contents of a second frame changes based on the user's choice. By default, a link loads into the same frame from which it was selected. You can change this default behavior and target the destination of a link to another frame in the frameset. To target from one frame to another, you first must name your frames using the NAME attribute in the <FRAME> element, and then target links to display their content in the named frame.
- Use special target names to solve unique design problems, favoring _top over _blank whenever possible.
- Frameset display is affected by the base screen resolution you use to code your framed pages. Decide on the lowest common denominator screen resolution that will display the frameset effectively. You also can build framesets using a combination of fixed and variable frame widths to accommodate different screen resolutions.
- Test your work. Different browsers and screen resolutions may affect the look of your framesets.

REVIEW QUESTIONS

1. What is the main benefit of frames?
2. Why do framesets add to initial download time?
3. Name two ways frames potentially may confuse users.
4. Why do search engines have problems with framesets?
5. What are the two attributes you use to structure the look of a frameset?
6. How does the order of the <FRAME> element within the <FRAMESET> affect the display of the frameset?
7. What is the purpose of the <NOFRAMES> element?
8. What is the benefit of the NORESIZE attribute?
9. What is the default setting for the SCROLLING attribute?
10. List the correct attributes to the <FRAMESET> element that will remove the frame borders in both Netscape Navigator and Internet Explorer.
11. What are the two tasks you must perform to add targeting to a frameset?
12. Which element lets you set a default target for all links in a document?
13. List the four special target names.
14. Are target names case-sensitive?
15. Which special target name is the default behavior for links in a frameset?
16. What is the major drawback of the _blank special target name?
17. How can you control the display of content within a frameset?
18. What additional browser elements can affect the display of your frameset?
19. How can you accommodate frameset display at varying screen resolutions?

PROJECTS

1. Browse the Web for a mainstream Web site that you think effectively uses frames.
 - a. Test the site at different resolutions.
 - b. Test the site in at least two different browsers.
 - c. Navigate the site and judge the effectiveness of the navigation and content presentation.
 - d. Write a summary of your findings.
2. Browse the World Wide Web for a framed mainstream Web site that you think would benefit from not using frames.
 - a. Test the Web site at different resolutions.
 - b. Test the Web site in at least two different browsers.
 - c. Navigate the Web site and judge the effectiveness of the navigation and content presentation.
 - d. Write a design critique and suggest ways that the Web site could be redesigned without using frames.
3. Visit the *Principles of Web Design* Companion Web Site and download some of the sample frame content files.
 - a. Build a two-column frameset that contains a fixed left navigation frame and a variable right content frame.
 - b. Target all of the links in the left frame to the right content frame.
 - c. Restrict the user's ability to resize the frames.
 - d. Test your work in different browsers and at different resolutions.
4. Visit the *Principles of Web Design* Companion Web Site and download some of the sample frame content files.
 - a. Build a two-row frameset that contains a fixed top navigation frame and a variable bottom content frame.
 - b. Target all of the links in the top frame to the bottom content frame.
 - c. Completely remove the frame borders.
 - d. Test your work in different browsers and at different resolutions.
5. Visit the *Principles of Web Design* Companion Web Site and download some of the sample frame content files.
 - a. Build a simple two-row or two-column frameset, or use one of the framesets from Exercise 3 or 4. Add links to your navigation frame that point to live Web sites.
 - b. Add a special target name that will load the linked Web sites into a new browser window.
 - c. Test the links and view the browser's behavior.
 - d. Test the results in different browsers.

6. Visit the *Principles of Web Design* Companion Web Site and download some of the sample frame content files.
 - a. Build a simple two-row or two-column frameset, or use one of the framesets from Exercise 3 or 4. Add links to your navigation frame that point to live Web sites.
 - b. Add a special target name that will load the linked Web sites at the top-level of the existing browser window.
 - c. Test the links and view the browser's behavior.
 - d. Test the results in different browsers.

CASE STUDY

Determine whether frames will enhance the effectiveness of your Web site and presentation of your content. Write a design summary that states how your Web site would benefit from the use of frames, and include a sketch of the structure and navigation of your proposed framed Web site. Discuss targeting behavior and how you would handle links to sites outside of your own. Discuss whether you would have a framed page or standard HTML page for the top-level page of your Web site.

If you determine that your Web site would benefit from frames, build a test frameset. Include some sample content pages. Test your frameset at different resolutions and in different browsers. Code your frameset to work at both 640 x 480 and 800 x 600 resolutions. If the testing shows positive results, adopt the frameset for your completed Web site.